Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1-36 canceled.

Claim 37 (new): A method for encoding a sequence of digital data, the method comprising:

storing a first data packet, the first data packet including a first sequential number and a first block identifier;

storing a second data packet, the second data packet including a second sequential number and a block width;

storing a third data packet, the third data packet including a third sequential number and a second block identifier;

storing a fourth data packet, the fourth data packet including a fourth sequential number and the block width;

transmitting the first data packet at a first time;

transmitting the second data packet at a second time, the second time being subsequent to the first time;

transmitting the third data packet at a third time, the third time being subsequent to the second time; and

transmitting the fourth data packet at a fourth time, the fourth time being one of subsequent to the third time and prior to the first time, wherein a data block corresponds to a portion of the sequence of digital data, the data block including the first data packet, the second data packet, the third data packet, and the fourth data packet.

Claim 38 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the sequence of digital data contains a sequence of progressive data.

Claim 39 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the sequence of digital data contains a sequence of digital image data.

Claim 40 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the data block contains redundancy information.

Claim 41 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein a start and an end of the data block are determined via at least one of the first block identifier and the second block identifier.

Claim 42 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein every n-th data packet includes at least one of the first block identifier, the second block identifier, and a third block identifier.

Claim 43 (new): A method for encoding a sequence of digital data as claimed in claim 42, wherein the remaining data packets include the block width.

Claim 44 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the first data packet, the second data packet, the third data packet, and the fourth data packet are consecutive data packets.

Claim 45 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the data block is an interleaver block.

Claim 46 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein a sequence of data blocks is determined.

Claim 47 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein a sequence of data blocks is determined based on at least one of a time stamp and a serial number.

Claim 48 (new); A method for encoding a sequence of digital data as claimed in claim 37, wherein a Real-Time Transfer Protocol is used as protocol.

Claim 49 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the first block identifier is equal to the first sequential number.

Claim 50 (new): A method for encoding a sequence of digital data as claimed in claim 48, wherein the first block identifier is determined from the first sequential number of the Real-Time Transfer Protocol used.

Claim 51 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein an unequal error protection method is used.

Claim 52 (new): A method for encoding a sequence of digital data as claimed in claim 51, wherein the unequal error protection method used is a UXP method.

Claim 53 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the first block identifier is equal to the second block identifier.

Claim 54 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the first block identifier is different than the second block identifier.

Claim 55 (new): A method for encoding a sequence of digital data as claimed in claim 37, wherein the first block identifier is different than the first sequential number.

Claim 56 (new): A method for decoding a sequence of digital data, the method comprising:

receiving a first data packet at a first time;

receiving a second data packet at a second time, the second time being subsequent to the first time;

receiving a third data packet at a third time, the third time being subsequent to the second time;

receiving a fourth data packet at a fourth time, the fourth time being one of subsequent to the third time and prior to the first time;

storing the first data packet, the first data packet including a first sequential number and a first block identifier;

storing the second data packet, the second data packet including a second sequential number and a block width;

storing the third data packet, the third data packet including a third sequential number and a second block identifier; and

storing the fourth data packet, the fourth data packet including a fourth sequential number and the block width, wherein a data block corresponds to a portion of the sequence of digital data, the data block including the first data packet, the second data packet, the third data packet, and the fourth data packet.

Claim 57 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the sequence of digital data contains a sequence of progressive data.

Claim 58 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the sequence of digital data contains a sequence of digital image data.

Claim 59 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the data block contains redundancy information.

Claim 60 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein a start and an end of the data block are determined via at least one of the first block identifier and the second block identifier.

Claim 61 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein every n-th data packet includes at least one of the first block identifier, the second block identifier, and a third block identifier.

Claim 62 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the remaining data packets include the block width.

Claim 63 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the first data packet, the second data packet, the third data packet, and the fourth data packet are consecutive data packets.

Claim 64 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the data block is an interleaver block.

Claim 65 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein a sequence of data blocks is determined.

Claim 66 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein a sequence of data blocks is determined based on at least one of a time stamp and a serial number.

Claim 67 (new); A method for decoding a sequence of digital data as claimed in claim 56, wherein a Real-Time Transfer Protocol is used as protocol.

Claim 68 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the first block identifier is equal to the first sequential number.

Claim 69 (new): A method for decoding a sequence of digital data as claimed in claim 67, wherein the first block identifier is determined from the first sequential number of the Real-Time Transfer Protocol used.

Claim 70 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein an unequal error protection method is used.

Claim 71 (new): A method for decoding a sequence of digital data as claimed in claim 70, wherein the unequal error protection method used is a UXP method.

Claim 72 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the first block identifier is equal to the second block identifier.

Claim 73 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the first block identifier is different than the second block identifier.

Claim 74 (new): A method for decoding a sequence of digital data as claimed in claim 56, wherein the first block identifier is different than the first sequential number.

Claim 75 (new): A computer readable media storing software instructions to encode a sequence of digital data, the software instructions causing a computing device to:

store a first data packet, the first data packet including a first sequential number and a first block identifier;

store a second data packet, the second data packet including a second sequential number and a block width;

store a third data packet, the third data packet including a third sequential number and a second block identifier;

store a fourth data packet, the fourth data packet including a fourth sequential number and the block width;

transmit the first data packet at a first time;

transmit the second data packet at a second time, the second time being subsequent to the first time;

transmit the third data packet at a third time, the third time being subsequent to the second time; and

transmit the fourth data packet at a fourth time, the fourth time being one of subsequent to the third time and prior to the first time, wherein a data block corresponds to a portion of the sequence of digital data, the data block including the first data packet, the second data packet, the third data packet, and the fourth data packet.

Claim 76 (new): A computer readable media storing software instructions to decode a sequence of digital data, the software instructions causing a computing device to:

receive a first data packet at a first time;

receive a second data packet at a second time, the second time being subsequent to the first time;

receive a third data packet at a third time, the third time being subsequent to the second time;

receive a fourth data packet at a fourth time, the fourth time being one of subsequent to the third time and prior to the first time;

store the first data packet, the first data packet including a first sequential number and a first block identifier;

store the second data packet, the second data packet including a second sequential number and a block width;

store the third data packet, the third data packet including a third sequential number and a second block identifier; and

store the fourth data packet, the fourth data packet including a fourth sequential number and the block width, wherein a data block corresponds to a portion of the sequence of digital data, the data block including the first data packet, the second data packet, the third data packet, and the fourth data packet.